Red Hat Certificate System 8.0 Release Notes

with Updates for Errata RHSA-2010:0838
Edition 8.0.7

Landmann

Red Hat Certificate System 8.0 Release Notes

with Updates for Errata RHSA-2010:0838
Edition 8.0.7

Landmann
rlandmann@redhat.com

Legal Notice

Abstract

Release Notes for Red Hat Certificate System 8.0

Table of Contents

These release notes contain important information related to Red Hat Certificate System 8.0 that may not be currently available in the Product Manuals. New features, system requirements, installation notes, known problems, resources, and other current issues are addressed here. You should read these Release Notes in their entirety before deploying Red Hat Certificate System 8.0.

1. New Features for Red Hat Certificate System 8.0

Red Hat Certificate System 8.0 is a major release of Certificate System, and many new, contemporary features have been added and existing features have been made more robust and flexible.

1.1. Certificate Renewal

Certificate renewal for all Certificate System-issued certificates has been reintroduced using the new profile framework. There are a number of new profiles to use for renewal, including encryption and signing certificates for both standard use and on tokens, and server certificate renewal. New inputs have been added to manage certificate renewal, so corresponding renewal profiles can be created for custom enrollment profiles.

1.2. Improved Subsystem Cloning

Cloning has been enhanced with *distributed numeric assignments* logic so that cloned CAs can efficiently divide and use serial numbers for certificates without becoming blocked because of inadequate serial number ranges.

1.3. Stronger SELinux Policies

SELinux policies are now required for every subsystem and run in enforcing mode by default, providing much more protection for Certificate System processes.

1.4. Improved UTF8 Support

The CA, OCSP, and DRM subsystems fully accept and interpret certificate requests generated using UTF-8 characters, both in the console and in the agent services pages. This support is for specific fields.

End users can submit certificate requests with UTF-8 characters in those fields and end users and agents can search for and retrieve certificates and CRLs in the CA and retrieve keys in the DRM when using those field values as the search parameters.

Four fields fully-support UTF-8 characters:

- Common name (used in the subject name of the certificate)
- Organizational unit (used in the subject name of the certificate)
- Requester name
- Additional notes (comments appended by the agent to the certificate)

NOTE

This support does not include supporting internationalized domain names, like in email addresses.

1.5. Enhanced Support for Third-Party ECC Modules

Certificate System 8.0, although it does not ship with an ECC module, does support loading and using third-party ECC PKCS#11 modules with the CA. The console can handle ECC-based SSL sessions, and the server generates and supports ECC certificates.

1.6. Simplified Signed Audit Logging

Audit log signing certificates are now created with all of the other default subsystem certificates as soon as a CA, DRM, OCSP, TKS, or TPS subsystem is configured. The log is also already configured and can be very easily enabled. Signed audit logs can be verified by auditors using the included **`AuditVerify`** script.

1.7. New Windows Smart Card Login Profile for Tokens

A new example profile is included with the regular CA profiles list which enabled the CA and TPS to issue certificates and enroll tokens that can be used to log into Windows systems.

1.8. Enhanced Security Officer Mode and Enterprise Security Client Configuration

Setting up and using security officer workstation has been improved and additional parameters have been added to the **`esc-pref.js`** configuration file to make configuring the Enterprise Security Client security officer settings easier and more flexible.

1.9. Expanded TPS Roles

A new role, the operator role has been added to the TPS subsystem. This role can view and search all tokens, certificates, and activities within the Token Processing System (TPS) but cannot edit any entries.

Additionally, the administrator role interface has been enhanced to allow administrators to create and edit users, assign profiles, and delete users directly.

1.10. Added IPv6 Support

The Certificate System 8.0 services can accept requests from all supported browsers, from other Certificate System subsystems, and from the administrative console over IPv6. The server also supports using IPv6 addresses in the Subject Alt Names of certificates, with certificate extensions, and with Certificate System scripts and tools.

1.11. Using HTTP1.1 for Publishing CRLs

HTTP 1.1 has been added as a supported protocol to use to publish CRLs, in addition to publishing to file and to LDAP. This makes publishing CRLs safer and more efficient, since "chunks" of CRLs can be published rather the entire CRL. If CRL publishing is ever interrupted, the process can resume smoothly.

1.12. Enhanced Installation Scripts

Certificate System creates and configures additional instances using the **`pkicreate`** script. An additional script, **`pkisilent`**, can be used to create and configure multiple subsystem instances quickly and without unnecessary user interaction. Both of these scripts have been enhanced and strengthened for changes to port separation, security domain configuration, and other updates to the structure of Certificate System subsystems.

2. Important Configuration Changes

There have been some significant changes to the structure and configuration of the Certificate System 8.0 installation, which are not directly related to new features in Certificate System 8.0.

2.1. Default Port Separation

Starting in Certificate System 8.0, there are three SSL ports, one each for each of the user interfaces (agents, administrators, and end entities). The web application folders are also separated, so each web service is independent and secure. The **`pkicreate`** script has been updated to permit both separated and non-separated port configurations.

The original RA and TPS standard and SSL ports remain the same, but new SSL ports have been added for end entities.

NOTE

Port separation was originally introduced in an update to Certificate System 7.3, but the default for this errata was still to use a single SSL port at installation. In Certificate System 8.0, the default configuration is to have separate ports.

Table 1. New Port Assignments for Certificate System 8.0

Subsystem	Standard	End-Entity SSL	Agent SSL	Admin SSL	Tomcat
CA	9180	9444	9443	9445	9701
RA	12888	12890	12889	12889	
OCSP	11180		11443	11445	11701
DRM	10180		10443	10445	10701
TKS	13180		13443	13445	13701
TPS	7888	7890	7889	7889	

2.2. Changes in the Security Domain

In previous releases of Certificate System, the security domain was maintained in an XML file for the CA, **`domain.xml`**. In Certificate System 8.0, the security domain configuration has been moved to LDAP entries within the CA's LDAP entry.

2.3. Renamed Directory Paths

In previous releases of Red Hat Certificate System, the subsystem directories had the term **`rhpki`** in the name, such as **`/etc/rhpki-tps/CS.cfg`** and **`/usr/lib/rhpki/native-tools`**. All directories have been renamed **`pki`**, such as **`/etc/pki-tps/CS.cfg`**.

2.4. Replacing Policy Framework with Profile Framework

The old policy framework for managing certificates was deprecated in Certificate System 7.1 and was removed entirely for Certificate System 7.2, 7.3, and 8.0. Any certificate enrollments or other operations must be performed using the new profile framework.

2.5. Removed Bulk Issuance Tools

The bulk issuance tool, **`bulkissuance`**, is deprecated in Certificate System 8.0 because it worked with the deprecated policy framework, not the new profile framework. To perform bulk issuance of certificates, use the **`httpclient`** tool to post the certificate requests to the CA's agent interface. An additional tool,

pki_load, can be obtained from Red Hat support to send the certificate requests to **httpclient**.

3. Supported Platforms

This section covers the different server platforms, hardware, tokens, and software supported by Red Hat Certificate System 8.0.

3.1. Server Support

The Certificate System subsystems are supported on the following platforms:

- Red Hat Enterprise Linux 5.3 and later for x86
- Red Hat Enterprise Linux 5.3 and later for x86_64

3.1.1. Server Requirements

Table 2. Red Hat Enterprise Linux Server Requirements

Component	Details
CPU	Intel — 2.0 ZZ Pentium 4 or faster
RAM	1 GB (required)
Hard disk storage space	Total is approximately 5 GB Total transient space required during installation: 1 GB Hard disk storage space required for installation: Space required to set up, configure, and run the server: approximately 2 GB Additional space for database growth in pilot deployment: approximately 1 GB Total disk storage space for installation: approximately 1 GB

3.1.2. Red Hat Enterprise Linux Considerations

Before installing the Certificate System packages, ensure that the proper dependencies are installed on the Red Hat Enterprise Linux system.

The following package groups and packages must be installed on all Red Hat Enterprise Linux systems:

- gnome-desktop (package group)
- compat-arch-support (package group)
- web-server (package group)
- kernel-smp (package)
- e2fsprogs (package)
- firefox (package)

On 64-bit Red Hat Enterprise Linux platforms, ensure that the 64-bit (x86_64) **compat-libstdc++** libraries are installed, and not only the 32-bit (i386) libraries. To confirm this, run the following command as **root**:

```
rpm -qi compat-libstdc++ --queryformat '%{NAME}-%{VERSION}-
%{RELEASE}.%{ARCH}.rpm\n' | grep x86_64
```

Numerous libraries should be displayed.

3.2. Client Support

The Enterprise Security Client is supported on the following platforms:

- Apple Macintosh OS X 10.5.8 and higher (Leopard) (Power PC, Intel)
- Microsoft Windows Vista 32-bit
- Microsoft Windows Vista 64-bit
- Microsoft Windows XP 32-bit
- Microsoft Windows XP 64-bit
- Red Hat Enterprise Linux 5.3 x86
- Red Hat Enterprise Linux 5.3 x86_64

3.3. Supported Web Browsers

The services pages for the subsystems require a web browser that supports SSL. It is strongly recommended that users such as agents or administrators use Mozilla Firefox to access the agent services pages. Regular users should use Mozilla Firefox or Microsoft Internet Explorer.

NOTE

The only browser that is fully-supported for the HTML-based instance configuration wizard is Mozilla Firefox.

Table 3. Supported Web Browsers by Platform

Platform	Agent Services	End User Pages
Red Hat Enterprise Linux	Firefox 3.x	Firefox 3.x
Windows Vista	Firefox 2.x	Firefox 2.x Internet Explorer 7 and higher
Windows XP	Firefox 2.x	Firefox 2.x Internet Explorer 6 and higher
Mac OS 10.5.8 and higher	Agent services are not supported for Mac	Firefox 2.x

3.4. Supported Smart Cards

The Enterprise Security Client supports Global Platform 2.01-compliant smart cards and JavaCard 2.1 or higher.

The Certificate System subsystems have been tested using the following tokens:

- Gemalto TOP IM FIPS CY2 64K token, both as a smart card and GemPCKey USB form factor key
- Gemalto Cyberflex e-gate 32K token (Red Hat Enterprise Linux and Mac systems)
- Safenet 330J Java smart card

Smart card testing was conducted using the SCM SCR331 CCID reader.

The only card manager applet supported with Certificate System is the CoolKey applet which ships with Red Hat Enterprise Linux 5.3.

3.5. Supported HSM

Red Hat Certificate System supports the Safenet Chrysalis-IT LunaSA and nCipher netHSM 2000 hardware security modules (HSM) by default. The tested and supported versions are listed in Table 4, "Tested HSM Versions for Red Hat Certificate System 8.0". Other HSMs can be added by loading their libraries in the local machine and configuring the default configuration files after the Certificate System packages are installed, but before configuring the instances; this is described in the *Administrator's Guide*.

Table 4. Tested HSM Versions for Red Hat Certificate System 8.0

HSM	Firmware	Appliance Software	Client Software
Safenet Chrysalis-ITS LunaSA	4.5.2	3.2.4	3.2.4
nCipher netHSM 2000	2.33.60	11.10	

4. Installing Red Hat Certificate System Subsystems

The following sections contain information on the prerequisites and procedures for installing Certificate System subsystems, including basic information that you need to begin installing the packages.

Installing and configuring Certificate System 8.0 subsystems is described in more detail in the *Installation Guide*.

4.1. Installation Notes

- Packages are non-relocatable. The Red Hat Certificate System base packages can not be installed to a user-designated location.
- Remove any installed **`libsqlite`** RPM files before installing the RA. The **`sqlite`** RPM files that ship with RA cause conflicts with those files.

4.2. Install the Required JDK

Certificate System requires Sun JDK 1.6.0. This JDK must be installed separately.

The OpenJDK can be installed by using **`yum`** or by downloading the packages directly from http://openjdk.java.net/install/. For example:

```
yum install java-1.6.0-openjdk
```

After installing the JDK, run **`/usr/sbin/alternatives`** as **`root`** to insure that the proper JDK is available:

```
/usr/sbin/alternatives --config java

There are 3 programs which provide 'java'.

  Selection    Command
------------------------------------------------
   1           /usr/lib/jvm/jre-1.4.2-gcj/bin/java
 + 2           /usr/lib/jvm/jre-1.6.0-openjdk/bin/java
*  3           /usr/lib/jvm/jre-1.6.0-sun.x86_64/bin/java
```

4.3. Verifying Red Hat Directory Server

All subsystems require access to Red Hat Directory Server 8.1 on the local machine or a remote machine. The Directory Server can be installed on Red Hat Enterprise Linux 5.3 32-bit, Red Hat Enterprise Linux 5.3 64-bit, or Solaris 9 Sparc 64-bit.

Check that the Red Hat Directory Server is already installed. For example:

```
yum info redhat-ds
Installed Packages
Name        : redhat-ds
Arch        : x86_64
Version     : 8.1.0
Release     : 1.4.el5dsrv
Size        : 136M
Repo        : installed
...
```

Install Red Hat Directory Server 8.1, if a directory service is not already available. For example:

```
yum install redhat-ds
```

Installing Red Hat Directory Server is described in more detail in the *Red Hat Directory Server Installation Guide*.

4.4. Verifying Apache

Apache 2.x must be installed on Red Hat Enterprise Linux systems in order to install the TPS subsystem. Check that the appropriate version of Apache is installed.

```
yum info httpd
Installed Packages
Name   : httpd
Arch   : x86_64
Version: 2.2.3
Release: 1.4.el5
Size   : 2.9 M
Repo   : installed
...
```

Install Apache if it is not already available. For example:

```
yum install httpd
```

4.5. Installing mod_nss

Before installing the subsystem packages on Red Hat Enterprise Linux, first install or upgrade **mod_nss**. **mod_nss** is required for all Red Hat Certificate System packages, but is not included in the Red Hat Certificate System repositories, so make sure that the appropriate Red Hat Network channels are configured.

```
yum install mod_nss
```

4.6. Installing through yum

To install the subsystems on Red Hat Enterprise Linux 5 (32-bit), run a command like the following for each subsystem:

```
yum install pki-subsystem
```

subsystem can be any of the Certificate System subsystems:

- **ca** for the Certificate Manager.
- **ra** for the Registration Authority.
- **drm** for the Data Recovery Manager.
- **ocsp** for the Online Certificate Status Protocol Responder.
- **tks** for the Token Key System.
- **tps** for the Token Processing System.
- **console** for the Java console.

When the installation process is complete, a URL to access this instance is printed to the screen which gives the subsystem instances hostname, port, and a login PIN to access the configuration wizard.

```
Configuration Wizard listening on http://hostname.domainname:unsecure-
port/subsystem_type/admin/console/config/login?pin=pin
```

For example:

```
http://server.example.com:9180/ca/admin/console/config/login?
pin=Yc6EuvuY2OeezKeX7REk
```

4.7. Installing from an ISO

Red Hat Certificate System 8.0 can also be downloaded from Red Hat Network as an ISO image. This ISO image contains an **RPMS/** directory which can be used as a local yum repository.

Place that **RPMS/** directory on a web server and then configure yum to use that location as a repository. After that, install Certificate System as described in Section 4.6, “Installing through yum”.

5. Documentation for Certificate System 8.0

The Red Hat Certificate System 8.0 documentation includes a complete set of usage and management documentation for both regular users and administrators. Along with the existing documentation set, there are important changes and enhancements to the 8.0 documentation:

5.1. Documentation Changes in 8.0

- The *Administrator's Guide* has been reorganized and partially rewritten to have a better structure and flow to the content. The intent of rewriting the *Administrator's Guide* is to make information easier and more intuitive to find.
- A new *Installation Guide* has been added to the doc set. This is based on the installation sections from the *Administrator's Guide*.
- A new *Certificate System Deployment Guide* has been written to cover PKI concepts and deployment planning.
- A new end-entities guide, *Using End User Services*, has been created to have a small, handy guide for the end-user services for the CA and RA which are available through Certificate System.

All of the new features implemented in Certificate System 8.0 are covered in the documentation:

- New information on port separation has been added in all of the guides and all examples and screenshots have been updated with the new port settings.
- The renewal sections in the *Administrator's Guide* have been rewritten and updated for the new profile framework. This includes adding information on new CA profiles for renewal and new procedures to renew user and server SSL certificates. The enrollment pages list in the *Agent's Guide* has also been updated.
- The existing auto enrollment proxy information has been added to the *Administrator's Guide*.
- A new method for publishing CRLs over HTTP has been added, and the corresponding sections of the publishing chapter in the *Administrator's Guide* have been updated.
- The new TPS operator role has been added to the TPS chapter of the *Agent's Guide*, and the information for the agent and admin roles has been updated.
- The cloning sections have been updated to cover enhancements for managing and assigning serial numbers and for changes in the configuration procedure.
- There is enhanced UTF-8 support for subject alt names in certificates. This has been noted in the *Administrator's Guide*.
- Some information on audit log signing has been added to the logs section of the *Administrator's Guide*.
- The procedure for loading third-party ECC modules to provide ECC support has been added to the *Installation Guide*.
- The procedure for enabling Windows smart card logins has been added to the token management chapter of the *Administrator's Guide*.
- Changes to the Enterprise Security Client configuration and to the security officer configuration have been added to the *Managing Smart Cards with the Enterprise Security Client* guide.
- The section on security domains has been updated to reference the new LDAP entries rather than the **`domain.xml`** file.
- The directory paths have been updated for the new locations.

5.2. Documentation with 8.0

Along with these release notes, the documentation for Certificate System includes the following guides:

- *Certificate System Deployment Guide* describes basic PKI concepts and gives an overview of the planning process for setting up Certificate System.

  This manual is intended for Certificate System administrators.

- *Certificate System Installation Guide* covers the installation process for all Certificate System subsystems.

  This manual is intended for Certificate System administrators.
- *Certificate System Administrator's Guide* explains all administrative functions for the Certificate System. Administrators maintain the subsystems themselves, so this manual details backend configuration for certificate profiles, publishing, and issuing certificates and CRLs. It also covers managing subsystem settings like port numbers, users, and subsystem certificates.

  This manual is intended for Certificate System administrators.
- *Certificate System Agent's Guide* describes how agents — users responsible for processing certificate requests and managing other aspects of certificate management — can use the Certificate System subsystems web services pages to process certificate requests, key recovery, OCSP requests and CRLs, and other functions.

  This manual is intended for Certificate System agents.
- *Managing Smart Cards with the Enterprise Security Client* explains how to install, configure, and use the Enterprise Security Client, the user client application for managing smart cards, user certificates, and user keys.

  This manual is intended for Certificate System administrators, agents, privileged users (such as security officers), and regular end users.
- *Using End User Services* is a quick overview of the end-user services in Certificate System, a simple way for users to learn how to access Certificate System services.

  This manual is intended for regular end users.
- *Certificate System Command-Line Tools Guide* covers the command-line scripts supplied with Red Hat Certificate System.

  This manual is intended for Certificate System administrators.
- *Certificate System Migration Guide* covers version-specific procedures for migrating from older versions of Certificate System to Red Hat Certificate System 8.0.

  This manual is intended for Certificate System administrators.

All of the latest information about Red Hat Certificate System and both current and archived documentation is available at https://www.redhat.com/docs/manuals/cert-system.

6. Bugs Fixed in Certificate System 8.0

Along with the many new features and enhancements in Red Hat Certificate System 8.0, this release is also a bug fixing and maintenance release.

The following bugs have been fixed in the 8.0 release of Red Hat Certificate System.

Table 5. Fixed Bugs

Bug Number	Description
209213	There was a random error in the Enterprise Security Client that when an enrolled card was inserted and removed, the **`Reset Password`** window would not allow users to type in a new password.
211053	There Enterprise Security Client would not restart on Microsoft Windows Vista after the machine was rebooted, so the client would have to be started manually.
223309	When a CA was cloned, certain attributes that are required for the clone were not properly copied over, including attributes containing the CA certificates. This meant that the cloned CA could not function.
223367 224902	When a subsystem configuration failed or if a subsystem were uninstalled and then a new instance, with the same instance name was created, then the configuration would fail at the internal database configuration window because there was no way to overwrite the existing database with the same name. A new checkbox has been added that allows the new instance to reuse the database name.
224691	With two TPS instances on the same server, if one instance were restarted, then the other one would be stopped.
224765	When cloning a CA, the configuration wizard would claim that a new administrator certificate was generated and imported into the browser, even though no certificate was created.
224889	If a TPS was configured not to support server-side key generation, the DRM-related server-side key generation parameters were still added to the **`CS.cfg`** file with wrong values.
240083 241423	Inconsistently, expired certificates were included and published in new CRLs.
243921 490461	Publishing certificates and CRLs to a text file (file-based publishing) failed.
245661	Cloning a DRM on an HSM resulted in numerous errors, such as Java null pointer exceptions, authorization failures, and improper configuration for the instance state, request numbers, and serial numbers.
246252	A CA would accept a certificate request which was signed with an obviously bad signature because it only verified the certificate request itself.
250188	During subsystem configuration, the Authority Key Identifier extension was not be generated on the issued subsystem certificates.
251226	Opening the CA console threw a null pointer exception referencing the CMSAdmin class because of a misnamed jar file.
251569	The search time limit for end-entities pages set in the **`web.xml`** file was being inconsistently ignored, which could allow the LDAP

482935	process (ns-slapd) to reach 100% CPU.
253323	When using the Certicom PKCS #11 module with the OCSP, the OCSP failed to start because the OCSP signing certificate nickname was missing in the database token configuration.
253657	When attempting to enroll 64K smart cards in the TPS, key generation failed, causing the enrollment process to fail. This is resolved with the newest CoolKey applet.
285241	The Enterprise Security Client on Microsoft Windows and Apple MacIntosh would intermittently display only one certificate for a token in the **View Certificates** window, even though dual certificates (signing and encryption) had been enrolled.
286491	The OCSP signing certificate incorrectly was generated with the Basic Constraints value set to indicate it was a CA signing certificate (isCA=true).
357581	The Validity Constraints extension was not properly implemented in the profile framework so that the before and after grace periods were not honored.
435170	Going through the RA configuration wizard could blank out the **nss.conf** file that was created when the instance was created. The initial installation was run as **root**, but the configuration process was run as **pkiuser**, which caused the file permission and write problems.
435476	The DRM failed to start when self-tests were enabled.
437188	When a user requested a certificate through the CA's end-entities pages, the enrollment form always prompted for a key archival option and claimed to archive the new keys successfully, even though the profile was not configured for key archival and, in fact, no keys were archived.
439027	Certificates could not be imported into a subsystem instance's certificate database using the administrative console.
441896 442387 443657 480804	A number of different actions in the CA console would make the console freeze and hang, including: » Deleting certificates from the certificate database. » Changing the automated notifications configuration. » Editing CRL issuing points. » Changing the validity nesting requirement setting.
442239	The contents of CRL issuing points entry fields were not being shown in the console window, so none of the entry values that were typed when adding or editing CRL issuing points were visible.
443120	Trying to delete an imported CA certificate in the console returned the message *You're not authorized to perform that operation*, and the deletion failed.
443413	The notification email sent to an agent for a new certificate request had the wrong URL to view the request.
443417	Even if the requester entered his email when submitting a certificate request, the notification email sent to the agent always read *requestor email is VALUE UNKNOWN.*

445436	Searching for certificates through the **`Revoke Certificates`** page in the CA's agent services reported a bad search filter. The schema used for search filter generation have been updated to enhance the filtering options.
458499	The Unique Subject Name Constraint rejected requests with duplicate subject names even if the existing certificate was revoked or expired (when the duplicate name should have been allowed). Along with fixes to this error, a new parameter has been added to the constraint to allow administrators to set whether to allow duplicate subject names as long as the key usage bits are different.
463343	Key generation failed on the DRM when it was configured to use nCipher netHSM.
478909	In some situations, the internal LDAP database for a CA could run out of connections because of memory leaks associated with the operations to get and set serial numbers.
480143	SELinux errors at the time an instance was created could potentially prevent the configuration wizard from opening for the OCSP, TKS, or RA.
480825	The publisher ignored the encoding parameter and always published files in base-64 encoding.
481177	Normally, when a certificate is published to a file, and then the certificate is revoked, the publisher removes the published file for the revoked certificate. However, base-64 encoded files were not being unpublished by the publisher.
481790	If a value other than a UUID was set for the **`OtherName`** parameter for the Subject Alternative Name extension, than the subject alternative name was ignored and not included in the issued certificate.
483184	Attempting to add or register a custom authentication plug-in to the CA configuration through the console threw several different Java IO exception errors, and adding the plug-in failed.
490551	The key size is now selectable in end-entity forms, so the same profile can be used to issue both 1024-bit and 2048-bit certificates, for example. The fully range is 512 bits to 8192 bits, for RSA keys.
490782	The security officer token was reset whenever the Enterprise Security Client **`esc-prefs.js`** file was edited to go from security officer mode to regular mode. The security officer token was mistakenly being formatted when it was inserted to control a user token format operation.
490814	The format operation for a token updates its master key from the default to a new one. However, the default master key version is set in the TPS **`CS.cfg`**. Whenever master key changeover occurred, the formatted tokens could not be re-enrolled because their new master key version did not match the one in the TPS configuration. However, changing the master key version in the TPS configuration prevented new tokens (with the default master key) from being enrolled. New configuration parameters have been added to allow the TPS configuration to set both the default master key version and an

	indexed key version.
491000	Trying to format or re-enroll a formatted security officer token caused the Enterprise Security Client to throw error 28 on the format window because the command to revoke the existing certificates failed.
491185	All of the schema and elements implemented according to RFC 2256 have been updated to RFC 5280.
492180	Trying to enroll a temporary token for a security officer with a lost token failed with error 28, claiming that the connection to the CA was lost. The actual error was that key generation failed.
492189	A security officer token that was a temporarily lost state could be used to log into the security officer work station.
492361	An improved error message has been added to clarify why the enrollment operation was rejected when a user attempts to format a token which was enrolled to a different user.
499291	Installing the Enterprise Security Client on Microsoft Windows Vista was interrupted or failed because it attempted to call the **`eginstall.exe`** drivers, which are not shipped with Vista.
499439	The tokens activities display page was hard-coded to display only 25 entries, regardless of the actual number of records. This limit has been removed.
500698	If a CA, TKS, or TPS were installed on an HSM, the subsystem could crash after a large number of token operations (10,000 or more). A memory leak in the token operation exhausted the memory on the HSM.
501336	The Enterprise Security Client hung if a user attempted to re-enroll a token when the profile was set to reject re-enrollment operations (**`RE_ENROLL=NO`**).
502861	Attempting to enroll a certificate through the Signed CMC Authentication User Certificate Enrollment form always failed with an authorization error.
503045	Attempting to perform a CMC revocation through the end-entities services pages failed with a Java null pointer exception and an authorization error.
503783	If a certificate request was submitted without a requester email address, the request notification email sent to the CA agent gave the email address for the requester of the previous certificate.
508378	Attempting to connect to the internal database using LDAPS (LDAP over SSL) failed and caused the subsystem console to hang.

7. Errata Releases for Certificate System 8.0

The following errata have been issued for Red Hat Certificate System, fixing important security and performance issues. The complete list of errata issued for Red Hat Certificate System 8.0 is available through Red Hat Network.

Table 6. Errata Releases

Advisory	Description	Release Date
RHSA-2010-0837	This erratum introduces bug fixes and enhancements for SCEP operations. - CVE-2010-3868 — The certificate authority allowed unauthenticated users to request the one-time PIN in an SCEP request to be decrypted. An attacker able to sniff an SCEP request from a network device could request the certificate authority to decrypt the request, allowing them to obtain the one-time PIN. With this update, the certificate authority only handles decryption requests from authenticated registration authorities. - CVE-2010-3869 — The certificate authority allowed the one-time PIN used in SCEP requests to be re-used. An attacker possessing a valid SCEP enrollment one-time PIN could use it to generate an unlimited number of certificates. - CVE-2004-2761 — The certificate authority used the MD5 hash algorithm to sign all SCEP protocol responses. As MD5 is not collision resistant, an attacker could use this flaw to perform an MD5 chosen-prefix collision attack to generate attack-chosen output signed using the certificate authority's key. This update also adds additional configuration attributes to enable SCEP, set SSL connection settings, and set allowed and required	November 12, 2010

	encryption and hash algorithms.	
RHBA-2010-0701	This erratum introduces enhancements for certificate and token policy settings. - Bugzilla #609331. It was possible to change the status of a token marked as permanently lost or destroyed using the TPS administrative user interface. With this update, a configuration parameter was added to allow the administrator to define the possible state transitions available to tokens to prohibit insecure transitions. - Bugzilla #615814. The policyConstraintsCritical value was always treated as false. Even if the value was set to true, the value in any issued CA certificate was false. - Bugzilla #615827. The number of policy mappings used to be limited to five in the profile policies. With this update, the number of mappings was changed allow for more than five profile policies.	September 17, 2010
RHBA-2010:0448	This update supplies bug fixes to the Enterprise Security Client on Mac OS X 10.5.8 (and later) and re-introduces Mac support for the Enterprise Security Client. - Bugzilla #230764. When an enrolled smart card was inserted into the computer, the CoolKey TokenD (the token daemon) would not successfully launch. This daemon is what allows Apple's Safari browser and Apple Mail client to communicate with CoolKey smart cards. (BZ# - Bugzilla #236795. The	June 1, 2010

special security officer mode of esc did not function on Mac. Security officer mode allows designated users to perform in-person token enrollments, as added security.

- Bugzilla #239891. If a user deployed an enrolled CoolKey token to visit secure sites in Safari or used the token with Apple Mail to send encrypted mail, the CoolKey token daemon required the user to type in the token's PIN value every time the smart card was accessed because the PIN wasn't being cached properly.
- Bugzilla #369011. The CoolKey token daemon was confused with the pre-installed Apple token daemons when an enrolled token was inserted. This resulted in sporadic or missing access to the CoolKey smart card features when using Safari or Apple Mail.
- Bugzilla #369031. The previous release of esc for Mac OS X was not compatible with Mac OS X 10.5 Leopard. This meant that CoolKey smart cards could not be enrolled on Mac.
- Bugzilla #403571. The CoolKey PKCS#11 module, shipped with esc, had problems functioning fully on Mac OS X 10.5 Leopard. This resulted in problems using esc to manage tokens and problems actually using the cards for cryptographic operations.
- Bugzilla #455340. If a user wanted to enroll and deploy smart cards with 2048-bit keys, the standard

	cryptographic functions expected of smart cards were not possible. For instance, an encrypted email could not be sent with 2048-bit keys.	
RHBA-2010:0169	This update supplies a fix to a vulnerability in the TLS/SSL protocols that could allow a specific type of man-in-the-middle attack. This errata (and related configuration changes) make all Certificate System subsystems compliant with RFC 5746.	March 25, 2010
RHBA-2010:0097	This update addresses a problem (Bugzilla 557346) in marking the Name Constraints Extension as critical. When the criticality was marked "true" in the enrollment profile or if the agent marked the criticality as "true" when approving the request, the criticality always came up as "false."	February 11, 2010
RHBA-2009:1687	This release added support for Enterprise Security Client on 64-bit Windows servers. This release included a new configuration parameter for the Enterprise Security Client which allows administrators to set up a shared security database with common certificates that can be used when there are multiple Enterprise Security Client users on a single machine. This release also included fixes for these bugs: • Bugzilla 530633. The Enterprise Security Client on Windows didn't recognize when a Gemalto token was inserted. • Bugzilla 530482. With a poor network connection, a blank screen would sometimes pop-up instead of the token enrollment window. A more	December 21, 2009

	helpful error message has been added to the client. - Bugzilla 523568. Smart cards could not be enrolled using LDAP authentication when the passwords where stored using the password storage scheme.	
RHBA-2009:1665	These packages included an enhancement which allows a subsystem to be configured to prompt for subsystem passwords rather than reading them out of the plaintext password.conf file. New instances can have the password.conf file removed and prompt for all necessary passwords immediately; existing and fully-updated instances can be configured to prompt for passwords once the password.conf file is removed. A new watchdog process ensures that the instance can restart cleanly if a process is interrupted. This release also included fixes for these bugs: - Bugzilla 529280. TPS HTTP responses were not properly formatted according to RFC 2616. Chunked data were supposed to end with the sequence 0\r\n, but this was not included in the TPS responses. - Bugzilla 533510. If signed audit logging was enabled for the TPS, then it was not possible to start the TPS instance.	December 11, 2009
RHBA-2009:1602	This release added functionality to select signature digest algorithms (like SHA256 and SHA512) for RSA and ECC. This release also included fixes for these bugs: - Bugzilla 529945	November 25, 2009

	» Bugzilla 351162	
RHBA-2009:1596	This update addresses Bug 505682 - Allow configuration of NSS OCSP cache settings. New parameters are enabled to allow user-defined cache sizes, OCSP check times, and timeout periods for OCSP responses.	November 19, 2009
RHBA-2009:1443	This release had enhancements for ECC support, including extending support on Firefox for ECC enrollments and adding support for ECC POP. This release also included these bug fixes: » Bugzilla 512831 » Bugzilla 512828 » Bugzilla 513450 » Bugzilla 514093 » Bugzilla 514270 » Bugzilla 518431	September 14, 2009

8. Known Issues

8.1. Reconfiguring the Red Hat Certificate System Subsystems to Prevent a Potential TLS-Related Man-in-the-Middle Attack

Transport Layer Security (TLS) is a protocol which establishes a secure connection between a client and a server. Marsh Ray of PhoneFactor discovered a flaw in the TLS protocol itself which could allow an attack to insert plain text into an existing session during a TLS renegotiation operation.

The Educated Guesswork blog has a good description of this kind of attack at http://www.educatedguesswork.org/2009/11/understanding_the_tls_renegoti.html.

Either a client or a server may request a renegotiation of an existing TLS/SSL session (for instance, to renew session encryption keys or to use different cipher suite). When TLS/SSL is used to secure access to an HTTP service and a client attempts to access some protected resource, server-initiated renegotiation asks client to authenticate with a certificate.

However, the TLS/SSL protocols did not use any mechanism to verify that session peers do not change during the session renegotiation. Therefore, a man-in-the-middle attacker could use this flaw to open TLS/SSL connections to the server, send attacker-chosen request to the server, trigger the renegotiation (either by directly requesting it or by attempting to access protected resource, resulting in server-initiated renegotiation) and splice victim's initial connection attempt to an existing TLS/SSL session. Depending on the application-layer protocol, this may lead to attacker request being performed by the server as if authenticated using victim's credentials or using data from victim's request. After the renegotiation, attacker can no longer decrypt communication between the client and the victim, so this attack is also referred to as a "blind prefix injection" attack. Eric Rescorla's blog post "Understanding the

TLS Renegotiation Attack" provides additional details about this flaw.

In Certificate System, this kind of session renegotiation occurs if a user connects to an end-entity port that doesn't require client authentication, but then attempts to submit a certificate enrollment form for an enrollment profile that requires client authentication. The Certificate System server requests and then parses a client certificate for the user.

For both client-initiated and server-initiated renegotiation to be fixed, then both the client and server need to be updated to apply RFC 5746. which resolves the man-in-the-middle vulnerability. For the Certificate System subsystems, the resolution is supplied through Errata RHBA-2010:0169 and Errata RHBA-2010:0165, plus these configuration changes.

Certificate System supports several different clients:

- Certificate System and third-party RA subsystems (used by both regular users and SCEP services)
- TPS subsystems, which connect to the CA for token operations
- The Windows Autoenrollment Proxy
- Web browsers, which are used by users to connect to the CA's end-entities pages

Updating the system NSS packages on any system that hosts a Certificate System subsystem will take care of all subsystem communication. When the NSS packages are updated, the CA-RA and CA-TPS connections will use the new session renegotiation protocol and all of the operations will proceed as normal.

Additional configuration changes may need to be made for the Windows auto-enrollment proxy or third-party RAs if those systems aren't updated to use the new renegotiation protocol. Contact Red Hat support for information on what needs to be done for those clients.

It is unclear on when browser clients will have updates available and applied to use the new session renegotiation protocol. If these clients aren't updated, but the server is, then the connections to the subsystem server may fail.

NOTE

These changes are not required if all clients accessing Certificate Systems are upgraded to support RFC 5746.

Certificate System 8.0 uses the Red Hat Enterprise Linux 5 system NSS packages. Updated NSS packages for Red Hat Enterprise Linux 5 are available as part of Errata RHBA-2010:0165. Existing instances need to be reconfigured to add the new port, and direct requests to this port. Any new instances will automatically have these changes applied.

Procedure 1. For Existing CAs

1. Before making any edits to the CA configuration, back up the following files:
   - /var/lib/*instance_name*/webapps/ca/WEB-INF/web.xml
   - /var/lib/*instance_name*/web-apps.ee/ca/ee/ca/ProfileSelect.template
   - /var/lib/*instance_name*/conf/server.xml
   - /etc/init.d/*instance_name*
2. Since database changes are also required, back up the database.
3. Modify the **`server.xml`** file to add the new client authentication end-entities port.

a. At the top of the file, replace the PKI status definitions with the following section, with the correct hostname and ports. Replace all the lines with the exact excerpt because there are important spacing differences in the definitions.

```
<!-- DO NOT REMOVE - Begin PKI Status Definitions -->
<!--
Unsecure Port       = http://server.example.com:9180/ca/ee/ca
Secure Agent Port   = https://server.example.com:9443/ca/agent/ca
Secure EE Port      = https://server.example.com:9444/ca/ee/ca
Secure Admin Port   = https://server.example.com:9445/ca/services
EE Client Auth Port = https://server.example.com:9446/ca/eeca/ca
PKI Console Port    = pkiconsole https://server.example.com:9445/ca
Tomcat Port         = 9802 (for shutdown)
-->
<!-- DO NOT REMOVE - End PKI Status Definitions -->
```

b. Add a section for the new port. Make sure that the **`clientAuth`** value is set to **`true`**. (The port number and **`serverCertNickFile`** and **`passwordFile`** directives should all match your instance information.)

```
<!-- Port Separation:  EE Secure Client Auth Port Connector -->
<Connector name="EEClientAuth" port="9446" maxHttpHeaderSize="8192"
        maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
        enableLookups="false" disableUploadTimeout="true"
        acceptCount="100" scheme="https" secure="true"
        clientAuth="true" sslProtocol="SSL"
        sslOptions="ssl2=true,ssl3=true,tls=true"
        ssl2Ciphers="-SSL2_RC4_128_WITH_MD5,-
SSL2_RC4_128_EXPORT40_WITH_MD5,-SSL2_RC2_128_CBC_WITH_MD5,-
SSL2_RC2_128_CBC_EXPORT40_WITH_MD5,-SSL2_DES_64_CBC_WITH_MD5,-
SSL2_DES_192_EDE3_CBC_WITH_MD5"
        ssl3Ciphers="-SSL3_FORTEZZA_DMS_WITH_NULL_SHA,-
SSL3_FORTEZZA_DMS_WITH_RC4_128_SHA,+SSL3_RSA_WITH_RC4_128_SHA,-
SSL3_RSA_EXPORT_WITH_RC4_40_MD5,+SSL3_RSA_WITH_3DES_EDE_CBC_SHA,+SSL3_R
SA_WITH_DES_CBC_SHA,-SSL3_RSA_EXPORT_WITH_RC2_CBC_40_MD5,-
SSL3_FORTEZZA_DMS_WITH_FORTEZZA_CBC_SHA,-
SSL_RSA_FIPS_WITH_DES_CBC_SHA,+SSL_RSA_FIPS_WITH_3DES_EDE_CBC_SHA,-
SSL3_RSA_WITH_NULL_MD5,-TLS_RSA_EXPORT1024_WITH_RC4_56_SHA,-
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA,+TLS_ECDHE_ECDSA_WITH_AES_256_CBC_S
HA"
        tls3Ciphers="-SSL3_FORTEZZA_DMS_WITH_NULL_SHA,-
SSL3_FORTEZZA_DMS_WITH_RC4_128_SHA,+SSL3_RSA_WITH_RC4_128_SHA,-
SSL3_RSA_EXPORT_WITH_RC4_40_MD5,+SSL3_RSA_WITH_3DES_EDE_CBC_SHA,+SSL3_R
SA_WITH_DES_CBC_SHA,-SSL3_RSA_EXPORT_WITH_RC2_CBC_40_MD5,-
SSL3_FORTEZZA_DMS_WITH_FORTEZZA_CBC_SHA,-
SSL_RSA_FIPS_WITH_DES_CBC_SHA,+SSL_RSA_FIPS_WITH_3DES_EDE_CBC_SHA,-
SSL3_RSA_WITH_NULL_MD5,-TLS_RSA_EXPORT1024_WITH_RC4_56_SHA,-
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA,+TLS_ECDHE_ECDSA_WITH_AES_256_CBC_S
HA"

SSLImplementation="org.apache.tomcat.util.net.jss.JSSImplementation"
        serverCertNickFile="/var/lib/pki-
ca/conf/serverCertNick.conf"
        passwordFile="/var/lib/pki-ca/conf/password.conf"
        passwordClass="org.apache.tomcat.util.net.jss.PlainPasswordFile"
        certdbDir="/var/lib/pki-ca/alias"/>
```

4. Modify the **/etc/init.d/***instance_name* initialization script to read the new status definitions.

   a. At line 242, replace the following lines. Replace all the lines with the exact excerpt below because there are important differences in whitespace in the quoted strings.

   ```
   unsecure_port_statement="Unsecure Port       = "
   secure_agent_port_statement="Secure Agent Port   = "
   secure_ee_port_statement="Secure EE Port      = "
   secure_ee_client_auth_port_statement="EE Client Auth Port = "
   secure_admin_port_statement="Secure Admin Port   = "
   pki_console_port_statement="PKI Console Port    = "
   tomcat_port_statement="Tomcat Port         = "
   ```

   b. Modify the highlighted code at around line 280.

   ```
                                  head=`echo "$line" | cut -b1-22`
                                  if      [ "$head" ==
   "$unsecure_port_statement"     ] ||
                                          [ "$head" ==
   "$secure_agent_port_statement" ] ||
                                          [ "$head" ==
   "$secure_ee_port_statement"    ] ||
                                          [ "$head" ==
   "$secure_ee_client_auth_port_statement" ] ||
                                          [ "$head" ==
   "$secure_admin_port_statement" ] ||
                                          [ "$head" ==
   "$pki_console_port_statement"  ] ||
                                          [ "$head" ==
   "$tomcat_port_statement"       ] ; then
                                          echo "    $line"
                                          total_ports=`expr ${total_ports}
   + 1`
                                  fi
                          fi
             done

                 if [ ${total_ports} -eq 7 ] ; then
                          return 0
   ```

5. Open the **web.xml** file.

   ```
   vim /var/lib/instance_name/webapps/ca/WEB-INF/web.xml
   ```

6. Add the following servlet mappings for submitting profiles to the secure end-entities client authentication URL:

   ```
      <servlet-mapping>
         <servlet-name>  caProfileSubmitSSLClient  </servlet-name>
         <url-pattern>   /eeca/ca/profileSubmitSSLClient  </url-pattern>
      </servlet-mapping>

      <servlet-mapping>
         <servlet-name>  caGetCertFromRequest  </servlet-name>
         <url-pattern>   /eeca/ca/getCertFromRequest  </url-pattern>
      </servlet-mapping>
   ```

7. Edit the profile selection template to use the URL for the new secure end-entities client authentication services port. For example, assuming the default end-entities client authentication SSL port of 9446:

   ```
   vim /var/lib/instance_name/webapps/ca/ee/ca/ProfileSelect.template
   ... original ...
       uri = 'profileSubmitSSLClient';


   ... update ...
       uri =
   'https://server.example.com:9446/ca/eeca/ca/profileSubmitSSLClient';
   ```

8. The new port information needs to be added to security domain description of the subsystem, as stored in the database.

   a. Connect to the database and update the schema.

      ```
      /usr/lib/mozldap/ldapmodify -p db_port -h db_host -D "cn=Directory
      Manager" -w db_password

      dn: cn=schema
      changetype: modify
      add: attributeTypes
      attributeTypes: ( SecureEEClientAuthPort-oid NAME
      'SecureEEClientAuthPort'  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 SINGLE-
      VALUE X-ORIGIN 'user defined' )
      -
      dn:cn=schema
      changetype: modify
      delete: objectClasses
      objectClasses: ( pkiSubsystem-oid NAME 'pkiSubsystem' DESC 'CMS defined
      class' SUP top STRUCTURAL MUST ( cn $ Host $ SecurePort $ SubsystemName
      $ Clone ) MAY ( DomainManager $ SecureAgentPort $ SecureAdminPort  $
      UnSecurePort ) X-ORIGIN 'user defined' )
      -
      add: objectClasses
      objectClasses: ( pkiSubsystem-oid NAME 'pkiSubsystem' DESC 'CMS defined
      class' SUP top STRUCTURAL MUST ( cn $ Host $ SecurePort $ SubsystemName
      $ Clone ) MAY ( DomainManager $ SecureAgentPort $ SecureAdminPort
      $SecureEEClientAuthPort $ UnSecurePort ) X-ORIGIN 'user defined' )
      ^C
      ```

   b. Add the new port information to the security domain entry for this subsystem.

      ```
      /usr/lib/mozldap/ldapmodify -p db_port -h db_host -D "cn=Directory
      Manager" -w db_password

      dn: cn=hostname:admin_port,cn=CAList,ou=Security Domain,dc=basedn
      changetype: modify
      add: SecureEEClientAuthPort
      SecureEEClientAuthPort: new_port_number

      ^C
      ```

8.2. List of Known Issues in Red Hat Certificate System 8.0

These are known issues in the 8.0 release of Red Hat Certificate System. When available, workarounds are included.

Table 7. Known Issues

Bug Number	Description	Workaround

223299	If a TKS master key is generated on a SafeNet LunaSA HSM, server-side key generation fails with the following error in the TKS debug log:
	`"can't generate key encryption key"`
	A similar message also appears in the debug log if server-side key generation is turned on:
	`"TokenServlet: key encryption key generation failed for` ***`CUID`***`"`
	CUID is the card unique ID.
		Do not use LunaSA HSMs to generate keys for the TKS subsystem.

223343	When an nCipher HSM is used for a Certificate System instance, the **`nfast`** group needs to include the user ID of the Certificate System instance process. For example, since default Certificate System instances run as **`pkiuser`**, then the **`pkiuser`** group needs to be added as a member to the **`nfast`** group, if the Certificate System group has not already been added as a member.	Add the Certificate System user, such as **`pkiuser`**, as a member of the **`nfast`** group.
223391	If there are multiple enrollment operations using the tpsclient tool when server-side key generation is enabled in the TPS, then the DRM connection can time out before the TPS can generate the keys. The tool will then return the error *Failed to generate key on server. Please check DRM.*	Edit the TPS **`CS.cfg`** configuration file and increase the timeout period for the connection to the DRM by adding the following line: `conn.drm1.timeout=25`
224837	The configuration wizard is still available even after the subsystem instance configuration is complete.	
224994	CEP currently logs any authentication failures during enrollment to the system log. These should log to the audit log.	
233024	The auto enrollment proxy configuration is not added to everyone's profile. This is typically occurs when configuring the auto enrollment proxy on Windows child domains where the local administrator does not have permission to modify the **`cn=configuration`** tree in Active Directory. The simplest workaround is to use the **`Run as ..`** option to authenticate as the primary domain controller administrator and to then try to modify the **`cn=configuration`**. This relates to the **`Populate AD`** option in AEP.	
234884	The Phone Home UI pops up for both enrolled and uninitialized tokens on RHEL4 and MAC OS X, even	Type in the Phone Home URL and proceed.

	though the tokens contain Phone Home URLs.	
235150	The TKS sub-system start and stop scripts currently do not check that the package is installed before attempting to execute.	
236857	In the RA agent page, the RA attempts to retrieve revocation information for a certificate that the agent does not have the rights to see. This is not an issue at present and can be ignored.	
237050	There can be numerous *File does not exist* errors in the RA error logs. The administrator can safely ignore these error messages.	
237056	On the agent interface of the RA, the List Requests page displays the total number of certificate requests. On the List Certificates page, the corresponding information is missing. This will be fixed in the next release.	
237250	There is currently no facility for canceling certificate revocation. This will be added in the next release.	
237251	There is currently no option to add comments to a revocation request submitted through the RA. This is useful for agents if they are temporarily putting certificates on hold. This facility is currently only provided in the CA. It will be added to the RA in the next release.	

237305	The CA subsystem in Certificate System does not process SCEP requests that have been previously submitted. This can result in an error message similar to the following:
	`1706.http-9080-Processor24 -`
	`[20/Apr/2007:05:47:23 PDT] [20] [3] CEP`
	`Enrollment: Enrollment failed: user used`
	`duplicate transaction ID.`
		To avoid this situation, ensure that the Cisco router generates fresh sets of keys for SCEP enrollments.

237353	If the user clicks a link in the agent interface too fast and too many times, the server may return *Broken pipe: core_output_filter: writing data to the network* and terminate the SSL connection.	Re-authenticate to the agent interface.
238039	The Subject Alt Name extension in certificates that are issued using the caDirUserCert profile contain unsubstituted variables, such as **`$request.requestor_email$`**), if the profile request does not contain values available for substitution.	
238203	The TPS instance name is hard-coded in the **`CS.cfg`**. Because the instance name is hard-coded, the TPS looks for the configuration file in **`/var/lib/rhpki-tps/conf/CS.cfg`**.	If you create an instance with a name other than **`rhpki-tps`**, modify the **`/var/lib/`***tps-instance-name***`/cgi-bin/sow/cfg.pl`** file to remove the hard-coded instance name.

456701	The default signing algorithm used by the CA cannot be successfully changed in the CA configuration or when setting up the CA. The default is hard-coded to MD5withRSA.	
453051 483359	When trying to renew a subsystem certificate using the certificate wizard tool in the Java console (**`pkiconsole`**), the certificate renewal fails and the console throws a Java exception, such as *UNKNOWNEXCEPTION-java.util. MissingRessourceException: Can't find resource for bundle com.netscape.admin. certsrv.CMSAdminResources, key UNKNOWNEXCEPTION*. The console relied on the old policy framework to renew certificates, but the policy framework was replaced by a new profile framework in Certificate System 7.2. Therefore, the renewal feature in the console is broken. This is related to bug 499014.	Use the certificate wizard in the console to generate new certificates for the subsystem. Alternatively, use the CA's web services forms to renew the certificate or create a new renewal profile for the subsystem certificates.
454559	Attempting to connect to the Online Certificate Status Manager using **`wget`** or HTTP POST to send OCSP requests times out.	Use the **`OCSPClient`** tool to send status requests.
476096 489558	Due to a security concern, the Red Hat Directory Server Perl files on Sun Solaris platforms were moved from **`/opt/perl5x`** to **`/usr/lib/sparcv9/dirsec/perl5x`**. However, some Perl utilities includes with Certificate System are hard-coded to reference **`/opt/perl5x`**. This move can cause problems if users running Red Hat Certificate System upgrade their local Directory Server to Red Hat Directory Server 8.0 on the same machine.	Create symlinks to the new Perl directory. *ln -s /usr/lib/sparcv9/dirsrv/perl5x /opt/perl5x*
491438	If the TPS server is unavailable, then the Enterprise Security Client opens a blank screen in security officer mode rather than returning an error message that the server is unreachable.	If a blank screen appears when opening the Enterprise Security Client in security officer mode, try restarting the TPS server, and then restarting the Enterprise Security Client.
498299	The ***`tokendb.allowedTransitions`*** parameter in the TPS configuration sets the revocation states that a token can be assigned. For example, a token can go from a valid state to a permanently lost state. The ***`tokendb.allowedTransitions`*** parameter can be set to allow a transition from a state where the certificates are permanently revoked back to the active state. However, the TPS will not allow a token to go from a permanently revoked state back to active. Even though those operations appear to complete successfully, the certificates on that token are still revoked.	

499014	When trying to renew a DRM certificate using the certificate wizard tool in the Java console (**`pkiconsole`**), the certificate renewal fails and the DRM crashes. The console relied on the old policy framework to renew certificates, but the policy framework was replaced by a new profile framework in Certificate System 7.2. Therefore, the renewal feature in the console is broken. This is related to bug 453501.	Generate and install new subsystem certificates using the certificate wizard in the console, rather than attempting to renew existing certificates.
499052	If the configured OCSP responder in the RA or TPS **`nss.conf`** file is not the default responder, then NSS attempts to verify the OCSP signing certificate used by the OCSP, but it instead creates an infinite loop attempting to verify the certificate status against itself.	Make sure that any OCSP responder in the RA or TPS **`nss.conf`** file is the default, such as the CA's internal OCSP service.
499291	The e-gate drivers (**`eginstall.exe`**) would not install properly on Windows servers, which caused installing the Enterprise Security Client to fail on Windows. The e-gate drivers have been removed from the Windows Enterprise Security Client packages on Windows to allow the client to be installed.	e-gate tokens must be formatted on Red Hat Enterprise Linux or Mac systems, since the e-gate drivers are not available for the Enterprise Security Client on Windows.
501299	Token operations can cause a large number of unindexed searches to be returned in the instance's internal Directory Server logs. An unindexed search shows up in Directory Server access logs as **`notes=U`**. Unindexed searches are resource-intensive and can affect performance for the Directory Server. However, most of the unindexed searches returned for Certificate System token operations are improperly labeled index searches when they are really indexed VLV searches (related to Red Hat Directory Server bug 507460). The remainder of the unindexed searches still had very low etimes for the searches and should not significantly affect Certificate System performance.	
503641	Attempting to load the Certicom ECC module fails if SELinux is in enforcing mode, the default setting for Certificate System 8.0. **`modutil`**, the tool which is used to load ECC modules, requests text relocation permissions for Certicom's **`/usr/lib/libsbgse2.so`** library. This is not allowed by SELinux's enforcing mode.	SELinux can be configured to allow **`/usr/lib/libsbgse2.so`** to have text relocation permissions, which allows the ECC module to be successfully loaded. 1. Change the file context to **`textrel_shlib_t`**.

		`chcon -t textrel_shlib_t '/usr/lib/libsbgse2.so'`
		2. Then change the default file context files on the system so that the updated context is preserved even if the system is fully relabel.
		`semanage fcontext -a -t textrel_shlib_t '/usr/lib/libsbgse2.so'`
		3. Reload the ECC module; this should be successful.
		`modutil -dbdir /var/lib/pki-ca/alias/ -nocertdb -add certicom -libfile /usr/certicom/lib/libsbcpgse.so`

504013	Because of potential security risks, SCEP enrollment is disabled through the RA for Certificate System 8.0, and the corresponding enrollment forms have been removed.	
504088	The **`CRMFPopClient`** tool is used to submit a CRMF request to a CA, with proof of possession that the CA can verify. The CA then generates and, optionally, returns a certificate request or generates a request and archives the key (for DRM transport certificates). Running the **`CRMFPopClient`** tool to generate a transport certificate request for a DRM returns the error *java.io.FileNotFoundException* when submitting the CRMF request to a CA.	Use the CA's web interface to submit the CRMF transport certificate request.
509804	Installing or migrating instances on a Safenet Chrysalis-IT LunaSA HSM could fail. SSL connections from the subsystem begin failing after a short period of time and the connection could not be re-established.	Make sure that the following line must be added to the **`/etc/Chrystoki.conf`** configuration file:

		`Misc { NetscapeCustomize=1023 ; }`
		Additionally, these two lines must be removed:
		`AppIdMajor=2;`
		`AppIdMinor=4;`

511327	Trying to set up a TPS using a Safenet Chrysalis-IT LunaSA HSM fails with an error indicating that the password to access the HSM was incorrect or that the CA was unavailable.	Safenet Chrysalis-IT LunaSA HSM tokens cannot be used to set up the TPS.
512029	If the same HSM partition is used to multiple Certificate System subsystem instances, than the instance names cannot be used more than once, even if the instances are on different hosts. If a user tries to configure a new instance with the same name (including the default options) as an existing instance, then configuration will stall at key generation with an error that the certificate subject name already exists.	When using an HSM, always use unique instance names.
512493	Client authentication to the Java console fails in Red Hat Certificate System 8.0 because the console is unable to verify the client certificate required for authentication. This means that the console cannot be configured to run over SSL.	**IMPORTANT** If CA is configured for client authentication over the admin port and that CA is a security domain manager, then **no new PKI subsystems can be configured that use that CA for its security domain**. New PKI instances register themselves to the security domain CA over the admin port but without using client authentication. If the CA requires client authentication, then the registration attempt fails. 1. Stop the server. `service pki-ca stop`

2. Open the **`CS.cfg`** file and change the ***`authType`*** value to the client authentication setting.

   ```
   vim
   /var/lib/pki-
   ca/conf/CS.cfg

   authType=sslclien
   tauth
   ```

3. Open the **`server.xml`** file and change the ***`clientAuth`*** value to **`true`** for the admin port, in the admin connector entry.

   ```
   vim
   /var/lib/pki-
   ca/conf/server.x
   ml

   ....
   <Connector
   port="9445"
   maxHttpHeaderSiz
   e="8192"

   maxThreads="150"
   minSpareThreads="
   25"
   maxSpareThreads="
   75"

   enableLookups="fa
   lse"
   disableUploadTim
   eout="true"

   acceptCount="100"
   scheme="https"
   secure="true"

   clientAuth="true
   "
   sslProtocol="SSL"
   ```

4. Start the server.

   ```
   service pki-ca
   start
   ```

5. Configure the console.

a. Open the user's console directory.

   ```
   /user-directory/.redhat-idm-console
   ```

b. Create new security databases.

   ```
   certutil -N -d .
   ```

c. Export the administrator user certificate from your browser and save it to a **.p12** file, such as **/tmp/admin.p12**.

d. Copy the administrator user certificate **.p12** file to the console directory, and use **pk12util** to import it into the security databases.

   ```
   cp -p /tmp/admin.p12 /user-directory/.redhat-idm-console
   # pk12util -i ./admin.p12 -d /user-directory/.redhat-idm-console
   ```

e. Export the 64-bit blob of the issuing CA certificate from the browser and save it to a file like **ca.crt**.

f. Import the CA

		certificate from the base 64-blob associated with the admin user cert.
		`certutil -A -d . -n ca -t CT,C,C -i ./ca.crt`
		6. The next time you run **`pkiconsole`**, it prompts for you to supply the security database password and admin certificate to allow client authentication.
		`pkiconsole https://server.example.com:9445/ca`

513450	The CA is missing the configuration to support the Authority Information Access extension for CRLs.	This entry can be added manually to the CA **`CS.cfg`** file. 1. Stop the CA instance. `service pki-ca stop` 2. Add the extension to the file. For example:

```
vim /var/lib/pki-ca/conf/CS.cfg

ca.crl.MasterCRL.extension.AuthorityInformationAccess.accessLocation0=http://hostname:9180/ca/ocsp
ca.crl.MasterCRL.extension.AuthorityInformationAccess.accessLocationType0=URI
ca.crl.MasterCRL.extension.AuthorityInformationAccess.accessMethod0=ocsp
ca.crl.MasterCRL.extension.AuthorityInformationAccess.class=com.netscape.cms.crl.CMSAuthInfoAccessExtension
ca.crl.MasterCRL.extension.AuthorityInformationAccess.critical=false
ca.crl.MasterCRL.extension.AuthorityInformationAccess.enable=false
ca.crl.MasterCRL.extension.AuthorityInformationAccess.numberOfAccessDescriptions=1
ca.crl.MasterCRL.extension.AuthorityInformationAccess.type=CRLExtension
```

3. Start the CA instance again.

```
service pki-ca start
```

The Authority Information Access extension is described in the CRLs extension reference chapter in the *Certificate System*

		Administrator's Guide.

523568	On Windows XP and Vista systems, logging into the Enterprise Security Client using LDAP authentication can fail if the password is stored using the SSHA hash and has the exclamation point (!) or dollar sign ($) characters.	The exclamation point (!) and dollar sign ($) characters must be properly escaped for a user to bind successfully to the Enterprise Security Client.
		- For the dollar sign ($) character, escape the dollar sign *when the password is created*:
		`\$`
		Then, enter only the dollar sign ($) character when logging into the Enterprise Security Client.
		- For the exclamation point (!) character, escape the character when the password is created *and* when the password is entered to log into the Enterprise Security Client.
		`\!`

9. Copyright and Third-Party Acknowledgments

Red Hat Certificate System recognizes third-party contributions to portions of its servers and clients.

9.1. Copyrights for Portions of the Server

9.1.1. Apache Software Foundation

Red Hat Certificate System TPS subsystems require a locally-installed Apache 2.0.x HTTP server. Although a local copy of this server is generally installed as part of the operating system (with its corresponding license located in **/usr/share/doc/***httpd-version***/LICENSE**, the latest version of this server is available at http://httpd.apache.org.

Red Hat Certificate System CA, DRM, OCSP, and TKS subsystems use a locally-installed Tomcat 5.5 web server. Although an appropriate server is installed when any of these subsystems are installed, the latest version of this server is available at http://tomcat.apache.org.

Red Hat Certificate System uses many components made available from Apache.

- The XML project jars are **crimson.jar** and **xalan.jar**. These are available at http://xml.apache.org.
- The Tomcat project jar files are **servlet.jar** and **jakarta-naming.jar**. These are available at

http://jakarta.apache.org/tomcat/index.html.

9.1.2. Mozilla Foundation

Red Hat Certificate System uses version 4.2 of the Java™ Security Services (JSS) libraries from the Mozilla Project. If any problems are found in these specific libraries, the source code and build instructions for the latest version of and, potentially, the binary images for newer versions are available at http://www.mozilla.org/projects/security/pki/jss/index.html.

Red Hat Certificate System also uses version 4.6 of the Netscape Portable Runtime (NSPR) libraries from the Mozilla Project. If any problems are found in these specific libraries, the source code and build instructions for the latest version of these libraries and, potentially, the binary images for newer versions are available at http://www.mozilla.org/projects/nspr/index.html.

Additionally, Red Hat Certificate System uses version 3.11 of the Network Security Services (NSS) libraries from the Mozilla Project. If any problems are found in these specific libraries, the source code and build instructions for the latest version of these libraries and, potentially, the binary images for newer versions are available at http://www.mozilla.org/projects/security/pki/nss/index.html.

Red Hat Certificate System includes a set of compiled binaries (from NSS 3.11) of several tools from the Mozilla Project provided for the convenience of the user. This includes **`certutil`**, **`cmsutil`**, **`modutil`**, **`pk12util`**, **`signtool`**, **`signver`**, and **`ssltap`**. If any problems are found in these specific tools, the source code and build instructions for the latest version of this tool and, potentially, a binary image for other newer tools are available at

http://www.mozilla.org/projects/security/pki/nss/tools/index.html.

Red Hat Certificate System includes version 1.5 R3 of Rhino JavaScript for Java™. If any problems are found in this specific distribution, the source code and build instructions for the latest version and, potentially, a binary image are available at http://www.mozilla.org/rhino/index.html.

9.1.3. Red Hat

Red Hat Certificate System requires a complete Red Hat Directory Server 8.1 binary. The open source portion of Certificate System is available at the following URL:

https://rhn.redhat.com

9.2. Copyrights for Certificate System Clients

These are the copyrights and third-party acknowledgments for portions of Red Hat Certificate System 8.0 clients.

9.2.1. Mozilla Foundation

USE AND AVAILABILITY OF OPEN SOURCE CODE. Portions of the Product were created using source code governed by the Mozilla Public License (MPL). The source code for the portions of the Product governed by the MPL is available from http://www.mozilla.org under those licenses.

Red Hat Enterprise Security Client uses the latest version of the XULRunner cross-platform package. XULRunner is a Mozilla runtime package that can be used to bootstrap XUL+XPCOM applications that are as rich as Firefox and Thunderbird. If any problems are found in this specific distribution, the source code and build instructions for the latest versions and, potentially, a binary image are available at http://developer.mozilla.org/en/docs/XULRunner_1.8.0.1_Release_Notes.

Red Hat Enterprise Security Client also uses the Netscape Portable Runtime (NSPR) libraries from the Mozilla Project. If any problems are found in these specific libraries, the source code and build instructions for the latest version of these libraries and, potentially, binary images for newer versions are available at http://www.mozilla.org/projects/nspr/index.html.

Red Hat Enterprise Security Client also uses the Network Security Services (NSS) libraries from the Mozilla Project. If any problems are found in these specific libraries, the source code and build instructions for the latest version of these libraries and, potentially, binary images for newer versions are available at http://www.mozilla.org/projects/security/pki/nss/index.html.

9.2.2. e-gate Smart Card Drivers, Libraries, and Modules

- e-gate Smart Card Drivers for Windows 2000/XP Copyright 2002-2003 Schlumberger. All rights reserved.
- e-gate Smart Card Driver for Mac OS X Copyright 2003 by Chaskiel Grundman.

  Copyright 2003 by Philip Edelbrock.

  Significantly based on the Alladin etoken driver (the T=1 code is not needed): Copyright 2002 by Andreas Jellinghaus.

  Copyright 2002 by Olaf Kirch.

  See license terms below for rights on both parts.

  Some header files are from the pcsclite distribution: Copyright 1999 David Corcoran.
- e-gate Smart Card Drivers for Windows 2000/XP:

  Limited Warranty/ Exclusive Remedies. Schlumberger warrants to the benefit of Customer only, for a term of sixty (60) days from the date of acquisition of the e-gate Smart Card ("Warranty Term"), that if operated as directed under normal use and service, the Software will substantially perform the functions described in its applicable documentation. Schlumberger does not warrant that the Software will meet Customer's requirements or will operate in combinations that Customer may select for use, or that the operation of the Software will be uninterrupted or error-free, or that all Software errors will be corrected. Schlumberger's sole obligation and liability under this limited warranty shall be, at Schlumberger's option, to remedy any substantial non-performance of the Software to the functional descriptions set forth in its applicable documentation. If Schlumberger is unable to satisfy the foregoing limited warranty obligations during the Warranty Term, then Schlumberger shall, upon Customer's written request for termination of this Agreement, refund to Customer all sums paid to Schlumberger for the licensing of the Software hereunder. These are Customer's sole and exclusive remedies for any breach of warranty.

  WARRANTY DISCLAIMER. EXCEPT FOR THE EXPRESS LIMITED WARRANTY SET FORTH IN SECTION 5 ABOVE, THE SOFTWARE IS PROVIDED AS IS. SCHLUMBERGER AND ITS SUPPLIERS MAKE NO OTHER EXPRESS WARRANTIES. TO THE EXTENT AUTHORIZED BY APPLICABLE LAW, ALL OTHER WARRANTIES WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT, ARE SPECIFICALLY DISCLAIMED. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS AGREEMENT.

  Limitation of Liability. Schlumberger's cumulative liability to Customer, or any third party, for loss or damages resulting from any claim, demand or action arising out of or relating to this Agreement or use of the Software ("Damages"), shall not exceed the net amount paid to Schlumberger for the licensing of the Software, in this case, the cost of the single e-gate Smart Card. In no event shall Schlumberger or any Supplier be liable for any indirect, incidental, special consequential or exemplary damages of any character, including, without limitation, damages for lost profits, goodwill, work stoppage, computer failure and all other commercial damages.
- e-gate Smart Card Driver for Mac OS X:

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- The names of its contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

9.2.3. MUSCLE Drivers, Libraries, and Modules

- MUSCLE smart card middleware and applets

  Copyright 1999-2002 David Corcoran.

  Copyright 2002 Schlumberger Network Solution.

  All rights reserved.

- MUSCLE smart card middleware and applets:

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.